BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

As rescanning documents *will not* correct images, please do not report the images to the Image Problem Mailbox.

(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) Publication number: **0 683 451 A2**

(12) EUROPEAN PATENT APPLICATION

(21) Application number: **95106933.5**

(51) Int. Cl.[6]: **G06F 1/32**

(22) Date of filing: **08.05.95**

(30) Priority: **09.05.94 JP 95186/94**
20.06.94 JP 136953/94

(43) Date of publication of application:
22.11.95 Bulletin 95/47

(84) Designated Contracting States:
DE FR GB IT NL

(71) Applicant: **CANON KABUSHIKI KAISHA**
30-2, 3-chome, Shimomaruko,
Ohta-ku
Tokyo (JP)

(72) Inventor: **Sunakawa, Shinichi, c/o Canon Kabushiki Kaisha**
3-30-2, Shimomaruko,
Ohta-ku
Tokyo (JP)
Inventor: **Shimada, Kazutoshi, c/o Canon Kabushiki Kaisha**
3-30-2, Shimomaruko,
Ohta-ku
Tokyo (JP)
Inventor: **Tatsumi, Eisaku, c/o Canon Kabushiki Kaisha**
3-30-2, Shimomaruko,
Ohta-ku
Tokyo (JP)
Inventor: **Mori, Shigeki, c/o Canon Kabushiki Kaisha**
3-30-2, Shimomaruko,
Ohta-ku
Tokyo (JP)
Inventor: **Matsubayashi, Kazuhiro, c/o Canon Kabushiki Kaisha**
3-30-2, Shimomaruko,
Ohta-ku
Tokyo (JP)
Inventor: **Harada, Takashi, c/o Canon Kabushiki Kaisha**
3-30-2, Shimomaruko,
Ohta-ku
Tokyo (JP)
Inventor: **Nagasaki, Katsuhiko, c/o Canon Kabushiki Kaisha**
3-30-2, Shimomaruko,
Ohta-ku
Tokyo (JP)
Inventor: **Fukuda, Ryoji, c/o Canon Kabushiki Kaisha**
3-30-2, Shimomaruko,
Ohta-ku
Tokyo (JP)

(74) Representative: **Tiedtke, Harro, Dipl.-Ing. et al**
Patentanwaltsbüro
Tiedtke-Bühling-Kinne & Partner
Bavariaring 4
D-80336 München (DE)

(54) Power supply control method in multi-task environment.

(57) An information processing apparatus, which operates in a multi-task mode, calculates a total consumption power of devices used by each task, and assigns higher execution priority to a task having the largest consumption power, thereby shortening the execution time of the task having the largest consumption power, and suppressing the total consumption power of the apparatus. When a device is started upon switching of tasks, if the total consumption power exceeds the allowable power of the apparatus by a power consumed upon restarting of the device, the task is set in a waiting state until operations of other devices are completed, the consumption power is lowered, and it is ready to use the device by the task.

EP 0 683 451 A2

BACKGROUND OF THE INVENTION

The present invention relates to an information processing apparatus, which can time-divisionally switch and execute a plurality of applications or tasks, a processing method of the apparatus, and a power supply control method for the apparatus.

Some electronic apparatuses which incorporate microcomputers and the like have a so-called power-down function of shutting down a power supply or operating the apparatuses in a low-consumption power mode when a memory, a display device, and the like are not accessed for a predetermined time.

However, since the conventional power-down function enters a power-down mode independently of an application which is being currently executed when no access is made for a predetermined period of time, even a device which is not necessary for the currently executed application enters the power-down mode only after an elapse of the predetermined period of time. Therefore, the consumed power and delay time required before the beginning of the power-down mode are wasted.

As will be described later with reference to Fig. 19, some devices require higher power than in a normal mode upon transition to a low-power mode or transition from a low-power mode to a high-power mode. In this case, when a given device enters the low-power mode after an elapse of a predetermined period of time from the last access thereto, and resumes the high-power mode by an access immediately after the beginning of the low-power mode, the consumption power required when the device is kept in the high-power mode during this interval becomes lower than that required when the device is switched between the two power modes.

Conventionally, information processing apparatuses which can execute multi-task processing are known. Each of these information processing apparatuses can apparently execute a plurality of parallel processing operations by executing the multi-task processing for time-divisionally executing a plurality of processing operations. With this processing, a single user can simultaneously execute a plurality of applications. When a plurality of terminals are connected to such an information processing apparatus, a plurality of users can simultaneously use the information processing apparatus.

An operating system (OS) which executes the above-mentioned multi-task processing includes a processing unit (program) called a scheduler. The scheduler determines a task to be executed when a plurality of tasks are executable. Non-allocated tasks are queued and are not processed until the next allocation time. Such a scheduling method includes round robin scheduling, priority scheduling, and the like.

In the round robin scheduling, queued tasks are managed in the form of a first-in-first-out (FIFO) list and are executed in the order of the list. In this method, a task whose allocation time to a CPU has passed is added to the end of the list.

In the priority scheduling, the following method is realized.

In a method of determining priority in correspondence with the execution levels of tasks, if there are a user level at which a program created by a user operates and a privileged level at which the OS operates, a higher priority level is assigned to a task with the privileged level. Also, a method of assigning a higher priority level to a task with a strict limitation on its execution time such as a dynamic image reproduction task, a method of determining priority in correspondence with the qualifications of user IDs, a method of determining priority in correspondence with the ratio of the busy time in the allocation time to a CPU, and the like are known.

In general, tasks with the same priority levels are classified to some ranks, so that the priority scheduling is adopted to manage tasks in each rank, and the round robin scheduling is adopted to manage tasks in units of ranks.

On the other hand, in these information processing apparatuses, processing circuits must operate at high speeds to execute a plurality of processing operations parallel to each other. As a result, consumption power increases, and the temperatures of these circuits become very high during operation. Since such an information processing apparatus includes many I/O devices, a large amount of heat is also generated by these I/O devices. The heat causes operation errors of the apparatus, and deteriorates the reliability of the apparatus. For this reason, it is important to reduce consumption power to suppress heat generation.

A method of reducing consumption power realized in, e.g., a personal computer as an example of the information processing apparatus will be described below. In general, the following four methods are popularly adopted.

(1) The number of circuits to which DC power is applied is reduced.

(2) The operation voltage of each circuit is lowered.

(3) The operation frequency of each circuit is lowered.

(4) The operation of a circuit which is not being used is stopped.

Other features and advantages of the present invention will be apparent from the following description taken in conjunction with the accompanying drawings, in which like reference characters designate the same or similar parts throughout the figures thereof.

BRIEF DESCRIPTION OF THE DRAWINGS

Fig. 1 is a schematic block diagram showing the arrangement of an information processing apparatus according to an embodiment of the present invention;

Fig. 2 is a block diagram showing the arrangement of CPU peripheral circuits in the information processing apparatus of the embodiment shown in Fig. 1;

Fig. 3 is a block diagram showing the arrangement of a power supply unit to respective I/O units in the information processing apparatus of the embodiment shown in Fig. 1;

Figs. 4A and 4B are flow charts respectively showing device I/O control and interrupt control upon completion of time measurement by a timer in the information processing apparatus of the embodiment shown in Fig. 1;

Figs. 5A and 5B are timing charts for explaining a power-saving control state of the information processing apparatus of the embodiment shown in Fig. 1;

Fig. 6 is a view for explaining priority control of a task scheduler in the information processing apparatus of the embodiment shown in Fig. 1;

Fig. 7 is a view showing the format of a task control block in the information processing apparatus of the embodiment shown in Fig. 1;

Figs. 8A to 8C are tables showing the contents of the task control block in the information processing apparatus of the embodiment shown in Fig. 1;

Fig. 9 is a flow chart showing control of I/O device acquisition processing of tasks in an information processing apparatus according to the first embodiment of the present invention;

Fig. 10 is a flow chart showing control of I/O device release processing of tasks in the information processing apparatus according to the first embodiment of the present invention;

Fig. 11 is a view showing various basic parameters for scheduling used by a task scheduler in an information processing apparatus according to the second embodiment of the present invention;

Fig. 12 is a view showing the management areas of device drivers in the information processing apparatus according to the second embodiment of the present invention;

Fig. 13 is a flow chart showing start control of devices in the information processing apparatus according to the second embodiment of the present invention;

Fig. 14 is a flow chart showing control upon completion of measurement by a timer in the information processing apparatus according to the second embodiment of the present invention;

Fig. 15 is a flow chart showing restart control of a waiting device in the information processing apparatus according to the second embodiment of the present invention;

Fig. 16 is a functional block diagram showing an electronic apparatus according to the third embodiment of the present invention;

Fig. 17 is a block diagram showing the hardware arrangement of an electronic apparatus according to an embodiment of the present invention;

Fig. 18 is a timing chart showing the mode transition timings and changes in consumption power states in the third embodiment of the present invention;

Fig. 19 is a timing chart for explaining conventional mode transition timings;

Fig. 20 shows an example of an access time interval table of the third embodiment;

Fig. 21 shows an example of an access time table of the third embodiment;

Fig. 22 is a flow chart showing the updating processing of the access time interval table upon device access in the third embodiment of the present invention;

Fig. 23 is a flow chart showing the processing for determining if the control enters a mode L upon completion of an access to a device in the third embodiment of the present invention;

Fig. 24 shows an example of a table for storing the hysteresis of access times of an application A to an HDD in the fourth embodiment of the present invention;

Fig. 25 shows an example of an HDD access time interval table in the fifth embodiment of the present invention; and

Fig. 26 is a timing chart showing changes in consumption power states upon transitions of modes according to the sixth embodiment of the present invention.

The power supply arrangement of this embodiment will be described below with reference to Fig. 3.

A switching power supply 27 converts an AC voltage of a commercial power supply input via an AC plug 28 into voltages (e.g., +5 V for digital circuits, -24 V for driving the display, and +20 V for motor drivers of the respective drives) used in the main body.

In the information processing apparatus of this embodiment, operation power is independently supplied to each unit, and the power supply is turned on/off for each unit, thus saving power. In some units, a power-saving mode is realized by stopping internal clocks or lowering the clock frequency. The unit arrangement of this embodiment also includes a CPU•memory unit 30 constituted by the CPU 1, the CPU peripheral circuit 2, and the main memory 3, and the communication unit 36 constituted by the communication interface 15 in addition to the above-mentioned units.

Note that the power supply to the CPU•memory unit 30 cannot be turned on/off, and this unit is kept ON when the power switch is turned on. The display unit 31 includes the display controller 5 and the display 6. The FDD unit 32 includes the FDD controller 7 and the FDD 8. The HDD unit 33 includes the HDD controller 9 and the HDD 10, and the CD-ROM unit 34 includes the CD-ROM controller 11 and the CD-ROM drive 12. The printer unit 35 includes the printer controller 13 and the printer 14.

Power is supplied to these functional units via switches 21 to 26 included in the power supply controller 4. These switches 21 to 26 have a one-to-one correspondence with the above-mentioned units. When a corresponding switch is turned off, power supply to the unit is stopped, and its processing can also be stopped. The ON/OFF control of these switches is executed by the CPU 1 via the CPU peripheral circuit 2.

Note that these switches may comprise mechanoelectric elements such as electromagnetic relays, lead switches, or the like or may comprise semiconductor switches such as MOS-FETs.

Control of the power mode of the respective device units will be described below with reference to Figs. 4A and 4B.

In this embodiment, a time elapsed from the last access of each unit is measured. When the next access to the unit is not made within a predetermined period of time, it is determined that the unit is not being used, and the unit is switched to the power-saving mode. For this purpose, a control circuit of each unit comprises a timer for measuring the elapsed time. Upon completion of the measurement of a timer value set in each timer, the timer generates an interrupt signal to the CPU 1. When an access is made to the unit in the power saving mode, the control circuit of the unit detects this access, and performs processing for resuming a function of a portion which has been stopped so far.

In the above description, the control circuit of each unit comprises the timer for measuring the elapsed time. As another method, the timer 1002 (see Fig. 2) may be used. That is, the operating system periodically looks up the measured value of the timer 1002 in real time, and stores execution times corresponding to respective I/O units in the memory 3, thereby measuring the operation times of the respective units.

A processing sequence executed when a certain task which must access one of I/O units is given with the right of execution by the scheduler of the operating system, and accesses a certain I/O unit will be described in detail below.

The task scheduling method and its effect in the information processing apparatus of this embodiment will be described below with reference to Figs. 5A and 5B.

The operation of this embodiment will be explained below with reference to the timing charts in Figs. 5A and 5B.

Fig. 5A shows the relationship between the execution states of respective tasks and the operation modes (a normal operation mode and a power-saving mode) of an I/O unit exclusively used by a certain task when the task scheduling method of this embodiment is not used.

In this case, an operation state when there are three tasks (i.e., tasks A, B, and C) which are virtually simultaneously executed will be explained with reference to Fig. 5A. Fig. 5A exemplifies a case wherein the task C exclusively occupies a certain I/O unit (to be referred to as an I/O unit C hereinafter) until its processing is completed. Assume that a scheduler equally assigns the right of execution to these tasks in scheduling for executing these tasks. A task assigned with the right of execution occupies the CPU machine cycle, and executes predetermined processing.

An abscissa 900 in Fig. 5A represents the time base, and time elapses from the left to the right in Fig. 5A. Reference numerals 90, 91, and 92 denote the execution timings of the tasks A, B, and C. In the execution timings 90 to 92, each task is executed at the timing of high (H) level, and is queued at low (L) level. In this chart, an overhead time required for, e.g., the task switching time of the scheduler is not shown. Reference numeral 93 denotes a change in power mode of an I/O unit incorporated in the apparatus. When the power mode 93 is at H level, the I/O unit C is set in the normal operation mode for performing a normal operation, and continues to be in an operation waiting state when no access is made. When the

executed are controlled to be switched by the above-mentioned round robin scheduling.

Reference numerals 50 to 53 denote queue headers of the respective priority levels. The queue header 50 with priority level 4 corresponds to the highest priority, and the queue header 53 corresponds to the lowest priority (priority level 1). Queued tasks are coupled to each of the queue headers 50 to 53 of the respective priority levels in a list structure. Reference numerals 54 to 57 denote queued tasks connected to the ends at the respective priority levels.

Each task has a table called a task controller block (TCB), and the operation of each task of this embodiment is controlled in accordance with the TCB. The TCBs of the tasks are linked via a bidirectional list. Queuing of queued tasks used in scheduling is also realized by the bidirectional list of the TCBs.

The contents of the TCB will be described below with reference to Fig. 7.

Reference numeral 70 denotes a task ID which is set by the operating system to identify a task. Reference numeral 71 denotes TCB link data for interlinking to the TCBs of other tasks. Reference numeral 72 denotes an execution queue link for linking queued tasks. Reference numeral 73 denotes link data for controlling the semaphore among tasks. Reference numeral 74 denotes queue priority data of the task. Reference numeral 75 denotes task status representing the current task state. The task status 75 includes flags indicating enable/disable, suspend/in-execution, and the like of execution. Reference numeral 76 denotes a device list indicating devices acquired by the task itself and its sub task. Reference numeral 77 denotes a total consumption power of devices acquired by the task.

The detailed arrangement of the TCB will be described below with reference to Figs. 8A to 8C.

Fig. 8A shows the arrangement of the device list 76. The device list 76 stores device information acquired by each task, and is updated upon acquisition of a device. In Fig. 8A, reference numeral 80 denotes a device ID for identifying an acquired device. Reference numeral 81 denotes the consumption power of each device. Reference numeral 82 denotes the acquired time of a device. Reference numeral 83 denotes the task ID of a task which acquired a device. Note that a device acquired by a sub task is recorded in the device list of its originating task as well as its own device list.

Fig. 8B shows the storage state of the total consumption power of devices acquired by each task. Reference numeral 84 denotes a storage area of the total consumption power. This area stores a total of device consumption powers 81 in the device list.

Fig. 8C shows the storage state of variables for the priority level. The initial value of the priority level of each task is set by the operating system. The set priority level is corrected in correspondence with the total consumption power of devices used by the task, thereby determining a new priority level. This correction value is calculated based on the following formula:

$$\text{Corrected Priority} = \text{Setting Value} + \text{Power Correction Value} \quad (1)$$

Referring to Fig. 8C, reference numeral 85 denotes a priority level set by the operating system. Reference numeral 86 denotes a power correction value based on the total consumption power of devices acquired by the task. Reference numeral 87 denotes a corrected priority level, which is corrected based on formula (1).

〈Description of Processing Operation (Figs. 9 and 10)〉

The flow of processing of this embodiment will be described below with reference to flow charts.

In this embodiment, upon acquisition of the right of use of a device, the device list in each TCB is updated, and the total consumption power is calculated. Furthermore, the priority level is changed in correspondence with the total consumption power.

The processing of a device driver for acquiring each device will be explained below with reference to Fig. 9.

In step S100, device acquisition processing is performed. More specifically, various kinds of information are checked to check if a device to be accessed by a task is ready. In step S101, it is checked if a desired device can be acquired. If YES in step S101, the flow advances to step S102; otherwise, the processing ends. In step S102, the new device acquired in step S100 is added to the device list 76 in the TCB of the task, which issued the acquisition request. In step S103, the consumption power of the acquired device is added to the total consumption power 84. In step S104, the correction value of the priority level is calculated in accordance with the new total consumption power, and the calculation result is stored in the corrected priority level 87. The processing ends, and the flow returns to the processing step of a host processing program which called this routine.

In order to perform the above-mentioned processing, areas shown in Fig. 11 are allocated in the main memory 3.

A total power register 110 for recording a total consumption power of the apparatus is allocated in a common memory area 3. The consumption power of the entire apparatus is stored in real time when device drivers update the contents of the total power register 110 in correspondence with changes in status. Furthermore, a maximum allowable power register 111 is allocated to indicate power which can be supplied from the power supply of the apparatus. In addition, a device start queue list 112 is allocated to manage devices which are waiting for starting. The device start queue list 112 stores an ID 113 of a start queued device, a task ID 114 of a task which issued a restart request, an entry 115 of a restart routine, and the like.

In a memory area for each device driver, as shown in Fig. 12, a command buffer 120 for temporarily storing a command received in the power-saving mode, and a management area including an area 121 for storing the consumption power in a normal operation of a device, an area 122 for storing the consumption power upon start of a device, and the like are allocated in units of device drivers. Furthermore, in the management area of each device driver, a timer area 123 set with a timer value for measuring the time required until a device is stabilized after starting is allocated.

The processing of the second embodiment will be described below with reference to the flow charts in Figs. 13 to 15.

Fig. 13 is a flow chart showing the flow of I/O trap processing which is started when a task accesses a device in the power-saving mode.

In step S200, the contents of an access request to a desired device are stored in the command buffer 120. In step S201, the value of the total power register 110 indicating the total consumption power of the entire apparatus is loaded into the CPU 1. In step S202, the value of the area 122 indicating the consumption power upon starting of the device is added to the value of the total power register 110, and it is checked if the sum exceeds the value of the maximum (allowable) power register 111. If YES in step S202, the flow advances to step S206; otherwise, the flow advances to step S203.

In step S203, it is determined that the device can be started, and the value of the area 122 indicating the consumption power upon starting of the device is added to the value of the total power register 110. The flow then advances to step S204. In step S204, an instruction is supplied to the power supply controller 4 and a controller of the device, and a power supply voltage is supplied to the device. In step S205, a stabilization waiting timer of the device is reset, a time measurement value stored in the timer area 123 is set as a timer value, and the timer starts time measurement, thus ending processing.

On the other hand, in step S206, since it is determined that the device cannot be started, the task status 75 of the TCB of a task which issued the start request is switched to "event waiting" status. The flow advances to step S207, and the task is added to the device start queue list, thus ending processing.

Fig. 14 is a flow chart showing the flow of interrupt processing generated upon completion of the time measurement by the stabilization waiting timer of the device. Note that the time measurement of the timer may be achieved by either a hardware circuit or a software program.

In step S210, a command for a desired device is fetched from the command buffer 120. In step S211, device I/O processing is performed in accordance with the command fetched in step S210. In step S212, the value of the area 122 indicating the consumption power upon starting of the device is subtracted from the value of the total power register 110, and the value of the area 121 indicating the consumption power in a normal operation of the device is added to the difference. In step S213, the first task in the device start queue list 112 is read. In step S214, it is checked if devices are registered in the device start queue list 112. If YES in step S214, the flow advances to step S215; otherwise, the processing ends. In step S215, the restart routine of a device registered in the device start queue list 112 is called, and the processing ends.

Fig. 15 is a flow chart showing the flow of processing for restarting a device registered in the start queue list 112.

In step S220, the ID 113 and the like of the corresponding device are deleted from the device start queue list 112. In step S221, the value of the area 122 indicating the consumption power upon starting the device is added to the value of the total power register 110. In step S222, the task status is changed to "executable". In step S223, an instruction is supplied to the power supply controller 4 and a controller of the device, and a power supply voltage is supplied to the device. In step S224, the stabilization waiting timer of the device is reset, a value in the timer area 123 is set in the timer, and the timer starts time measurement, thus ending processing.

The above-mentioned processing can prevent the total consumption power of a plurality of devices from exceeding the maximum consumption power of the apparatus when these devices are simultaneously started.

By calculating the two formulas, it can be determined which one of powers with and without a transition is larger. The consumption powers P1, P2, P3, and P4 in the above-mentioned states, and the times (t2 - t1) and (t4 - t3) required for status transitions are defined in units of devices.

If an end trigger is generated at time t1, time t3 is predicted at time t1, and whether or not a transition to the mode L is to be made is determined based on which result of two formulas (3) and (4) is larger. If time t3 at which a use trigger is generated satisfies formula (5) below, it is determined that a transition to the mode L is to be made after generation of an end trigger in the mode H to obtain a smaller consumption power; otherwise, it is determined that a transition to the mode L is not to be made even after generation of an end trigger in the mode H to obtain a smaller consumption power.

From $(t3 - t2) \times (P2 - P1) > (P3 - P2) \times (t2 - t1) + (P4 - P2) \times (t4 - t3)$,

$$t3 > t2 + \{(P3 - P2) \times (t2 - t1) + (P4 - P2) \times (t4 - t3)\}/(P2 - P1) \quad (5)$$

As described above, in the third embodiment, by predicting time t3 at time t1, whether or not a transition from the mode H to L is to be made is determined.

Fig. 19 is a timing chart for explaining conventional status transition. In this case, a status transition from the mode H to L is not made immediately after an end trigger is generated, but is made after an elapse of a time (t1' - t1). The status transition from the mode H to L is completed within a time (t2' -t1'), and thereafter, a status transition from the mode L to H is made in response to a use trigger generated at time t3.

As can be seen from a comparison between Figs. 18 and 19, the time (t4 - t1) between the generation timings of an end trigger in the mode H and a use trigger remains the same, but the consumption power of the conventional control is considerably larger than that in the third embodiment.

A case will be exemplified below wherein the above-mentioned principle is applied to a multi-task system. Assuming that an application A accesses a certain device once per **a** seconds, an application B accesses the device once per **b** seconds, and an application C accesses the device once per **c** seconds, the device is accessed at a rate of $\{(1/\mathbf{a}) + (1/\mathbf{b}) + (1/\mathbf{c})\}$ times per second. In other words, the device is accessed once per $1/\{(1/\mathbf{a}) + (1/\mathbf{b}) + (1/\mathbf{c})\}$ seconds.

Therefore, time t3 at which a use trigger is generated can be predicted to be:

$$t3 = t1 + 1/\{(1/\mathbf{a}) + (1/\mathbf{b}) + (1/\mathbf{c})\} \quad (6)$$

By substituting time t3 in formula (5), whether or not a transition to the mode L is to be made is determined.

As shown in Fig. 20, time intervals (time elapsed from the previous access to the next access) of accesses made by applications to the respective devices such as the input unit 132, the display unit 134, the ROM 136, the RAM 137, the HDD 140, the communication unit 141, and the like are stored in the form of a table. The above-mentioned values **a**, **b**, **c**, and the like are determined in accordance with these access time intervals. The values in the table are updated by the following method every time applications (A to E) actually access the above-mentioned devices.

Fig. 21 shows the end times of previous accesses made by the respective applications to the respective devices. In Fig. 21, each end time is expressed by an absolute time (seconds) in accordance with the time (seconds) elapsed from the beginning of use of this system. In the following description, these times are similarly expressed.

Assuming that a certain application accesses a certain device, a value obtained by subtracting the time (the end time of the previous access) shown in Fig. 21 from the current time corresponds to the time interval until the current access. In order to more accurately predict an access time interval until the next access, not only the current time interval but also the previous time intervals are preferably used. For this reason, an average value between a time interval (a value obtained by subtracting the time in Fig. 21 from the current time) until an access to a certain device by a certain application, and a previous access time interval (the value shown in Fig. 20) to the device by the application is calculated, and the table value (Fig. 20) corresponding to the application and the device is updated using the calculated value as a new access time interval.

Upon completion of an access to a certain device by a certain application, the time value corresponding to the application and the device shown in Fig. 21 is updated to be the current time.

Fig. 16 is a functional block diagram of the third embodiment based on the above-mentioned functions.

Reference numeral 160 denotes a device control unit, which corresponds to the control circuits 133, 135, 138, 139, and 142 of the respective units shown in Fig. 17. Reference numeral 161 denotes a calculation/discrimination unit, which corresponds to the CPU 130, the main memory 131, and the like

In the third embodiment, upon completion of an access to a certain device, whether or not a transition to the low-power mode is to be made is determined without waiting for an elapse of a predetermined period of time. For this reason, the low-power mode can be efficiently set, and a power-down function can be realized.

[Fourth Embodiment]

In the fourth embodiment, a hysteresis of intervals between the previous accesses and the next accesses to a certain device is recorded. The consumption power W1 (t3) at time t3 in each of past n accesses is calculated using formula (3), and these values are averaged to obtain an expected value W3 of the consumption power.

$$W3 = \{\Sigma(P3 \times (t2 - t1) + P1 \times (t3 - t2) + P4 \times (t4 - t3))\}/n \quad (8)$$

where Σ is the sum of n consumption powers.

Formulas (4) and (8) are calculated, and whether or not a transition to the mode L is to be made is determined by comparing the calculated values. If W3 < W2, a transition is made, otherwise, a transition is not made.

In formula (8) above, since the values P1, P3, P4, (t2 - t1), and (t4 - t3) are defined in units of devices, a table for storing values (t3 - t2) (the time in the mode L) upon accesses to devices in correspondence with applications is prepared, as shown in Fig. 24.

In this manner, the expected values of the consumption powers with and without status transitions from the mode H to L are compared, and whether or not a transition is to be made is determined based on the comparison result. As a result, a possibility of selecting a transition with a smaller consumption power can be increased.

[Fifth Embodiment]

The hard disk drive (HDD) 140 is normally accessed as a file. Also, the HDD 140 is accessed as, e.g., a virtual memory for the purpose of saving the memory contents. Therefore, the access time interval of the HDD varies depending on states, i.e., if a file is open, if the virtual memory is enabled, and so on. Fig. 25 shows an example of access time intervals in such states.

As can be seen from Fig. 25, the access time interval of the HDD 140 when the virtual memory is ON is shorter than that when the virtual memory is OFF, and the access time interval when a file is open is considerably shorter than that when a file is closed.

Using Fig. 25 as a time interval table, whether or not a status transition is to be made is determined as in the third embodiment. Other devices may often be accessed as a file, and the same applies to these devices.

According to the fifth embodiment, the access time interval of a device can be predicted not only in units of applications but also in correspondence with the time intervals in units of system status.

[Sixth Embodiment]

In Fig. 18 described above, if the delay time required for the status transition from when a use trigger is input at time t3 until the device can be actually used at time t4 is long, a user must wait for a relatively long period of time, thus adversely influencing operability.

As shown in the timing chart in Fig. 26, an imaginary consumption power is added to the actual consumption power P4 in the transient state from the mode L to the mode H, and the sum is represented by P5. Whether or not a transition to the mode L is to be made is determined using formula (9) below in place of formula (5) above:

$$t3 > t2 + \{(P3 - P2) \times (t2 - t1) + (P5 - P2) \times (t4 - t3)\}/(P2 - P1) \quad (9)$$

Since P5 > P4, a possibility that inequality (9) is satisfied is lower than that for formula (5), and transitions to the mode L rarely take place.

The sixth embodiment is effective when a high-speed transition from the mode L to the mode H is required at the cost of a slight increase in consumption power.

7. The apparatus according to claim 6, characterised in that when the task begins to use a given device, said use power storage means reads out the consumption power upon starting of the device the use of which has begun from said storage means, and adds the readout consumption power to the total consumption power.

8. The apparatus according to anyone claim of claims 4 to 7, characterised by further comprising time measurement means (123) for measuring a time required for starting the device.

9. The apparatus according to claim 8, characterised in that when time measurement by said time measurement means ends, said use power storage means subtracts the consumption power upon starting of the device from the total consumption power, reads out the consumption power in a normal operation of the device from said storage means, and adds the readout consumption power to the total consumption power.

10. The apparatus according to anyone claim of claims 4 to 9, characterised in that when the total consumption power stored in said use power storage means becomes small, and it is ready to start a waiting task, said control means starts the waiting task.

11. A processing method in an information processing apparatus for time-divisionally executing a plurality of tasks, characterised by comprising the steps of:

calculating (S103) a total consumption power of devices accessed by each task;

changing priority levels of the tasks (S104) in accordance with the total consumption power of each task, so that higher priority is given to a task with a larger total consumption power; and

controlling an execution order of processing operations by the tasks in accordance with priority information which determines the priority levels of the tasks.

12. The method according to claim 11, characterised by further comprising the step of decreasing power (S122) to be supplied to a device which is not accessed for not less than a predetermined period of time.

13. The method according to claim 12, characterised by further comprising the step of canceling the decrease in power supplied to the device when a task issues an access request to the device to which the decreased power is supplied.

14. A processing method in an information processing apparatus for time-divisionally executing a plurality of tasks, characterised by comprising the steps of:

controlling an execution order of processing operations based on the tasks in accordance with priority information which determines priority levels of the tasks;

calculating and storing (S103) consumption powers of devices which may be used by a task, and a total consumption power of devices used by the task;

determining (S201) a total consumption power upon starting of a new device on the basis of the consumption power of the new device when the new device is used;

determining (S202) whether or not the determined total consumption power is not more than a maximum allowable power of said apparatus; and

permitting (S204) the use of the device by the task when it is determined that the determined total consumption power is not more than a maximum allowable power of said apparatus, and controlling (S206, S207) so processing by the task waits when it is determined that the determined total consumption power is more than a maximum allowable power of said apparatus.

15. The method according to claim 14, characterised by further comprising the step of subtracting (S112) the consumption power of a device the use of which has ended from the total consumption power when the use of the device by the task ends.

16. The method according to claim 14, characterised in that the consumption powers of the devices include a consumption power upon starting of each device and a consumption power in a normal operation of the device.

discriminating (S14) based on the prediction result whether or not the device is to be set in a low-power mode; and

operating the corresponding device in the low-power mode on the basis of the discrimination result.

29. The method according to claim 28, characterised in that the device includes at least one of an input unit, a display unit, and an external storage device.

30. The method according to claim 28, characterised by further comprising the step of storing an access end time to the device, and wherein the access time interval is calculated based on the stored end time, and the stored time interval is updated with reference to the stored time interval.

31. The method according to claim 28, characterised in that the discriminating step includes the step of comparing a first power which is saved until the predicted time, and a second power as an extra power due to a status transition to the low-power mode and a status transition from the low-power mode to a normal state until the next access, and discriminating that the low-power mode is to be set when the first power becomes larger than the second power.

32. The method according to claim 28, characterised in that the discrimination step includes the step of comparing a consumption power expected value until the next access, which is obtained when the low-power mode is set based on the predicted time, and a consumption power until the next access, which is obtained when the low-power mode is not set, and discriminating that the low-power mode is to be set when the consumption power is larger than the consumption power expected value.

FIG. 2

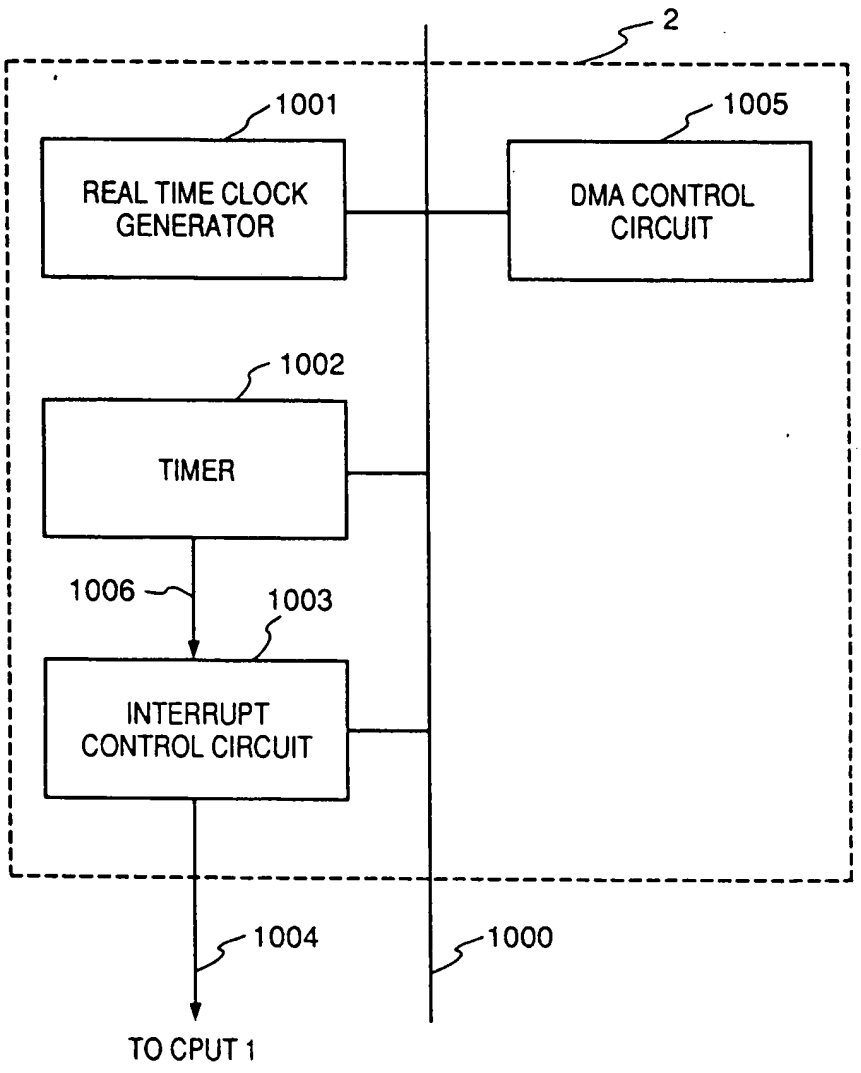
2
1001
REAL TIME CLOCK GENERATOR
1005
DMA CONTROL CIRCUIT
1002
TIMER
1006
1003
INTERRUPT CONTROL CIRCUIT
1004
1000
TO CPUT 1

FIG. 4A

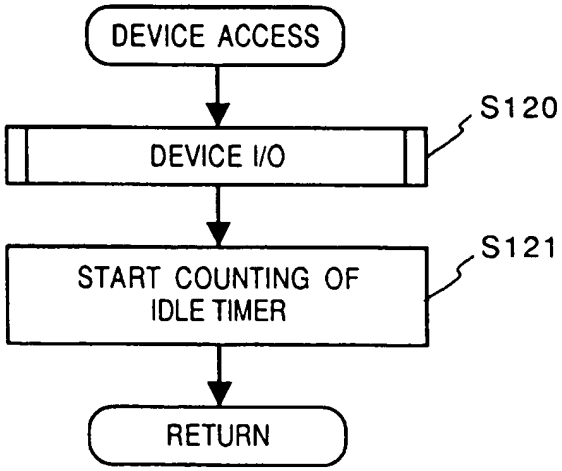
DEVICE ACCESS
DEVICE I/O
S120
START COUNTING OF IDLE TIMER
S121
RETURN

FIG. 4B

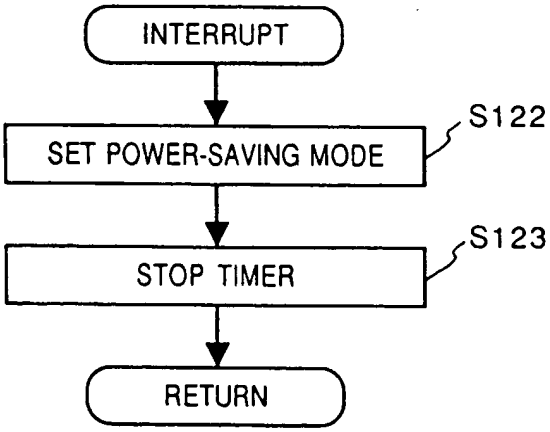
INTERRUPT
SET POWER-SAVING MODE
S122
STOP TIMER
S123
RETURN

FIG. 6

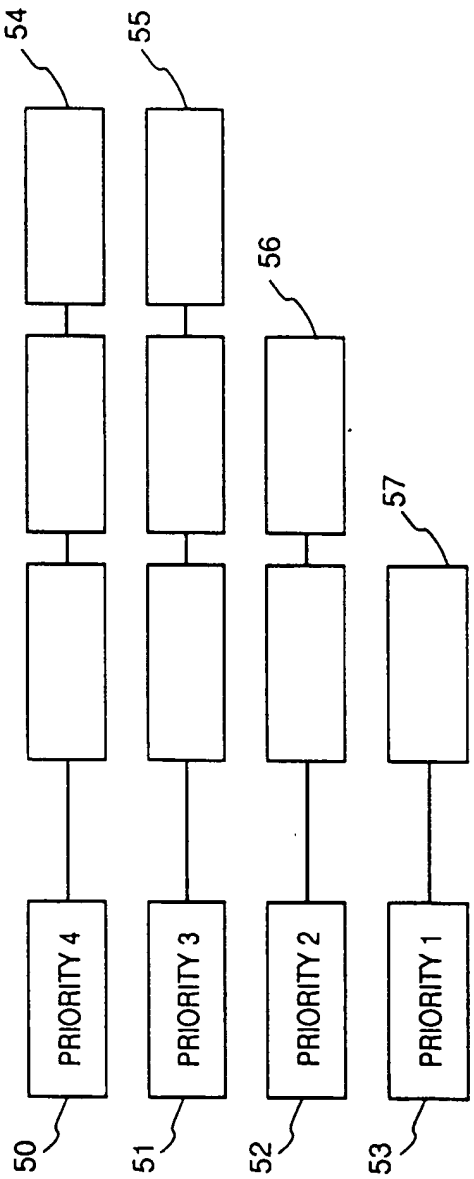
54
55
56
57
PRIORITY 4
PRIORITY 3
PRIORITY 2
PRIORITY 1
50
51
52
53

FIG. 8A

DEVICE ID	CONSUMPTION POWER	ACQUIRED TIME	TASK ID
0AH	80	11 : 30"30	ITSELF
10H	150	11 : 28"10	ITSELF
03H	500	11 : 31"45	30H
⋮	⋮	⋮	⋮

80 81 82 83

FIG. 8B

TOTAL CONSUMPTION POWER
730

84

FIG. 8C

SET PRIORITY	100
POWER CORRECTION VALUE	20
CORRECTED PRIORITY	120

85 86 87

FIG. 10

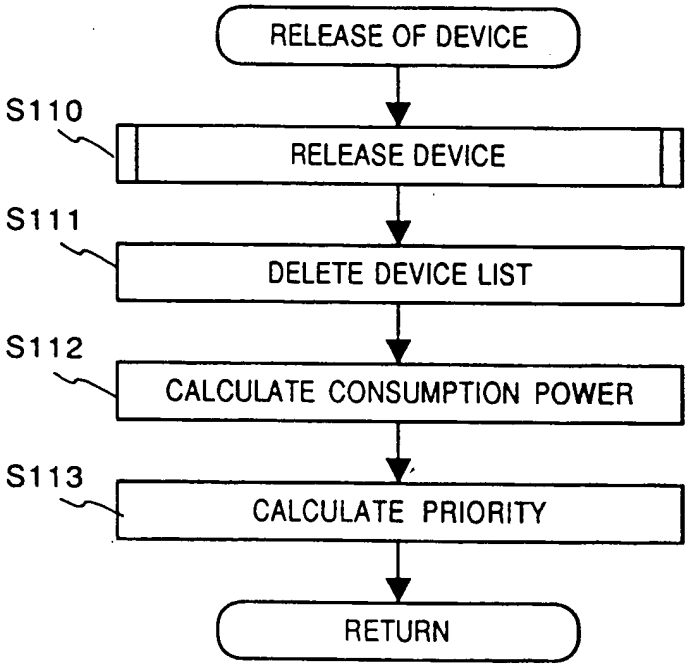
RELEASE OF DEVICE
S110
RELEASE DEVICE
S111
DELETE DEVICE LIST
S112
CALCULATE CONSUMPTION POWER
S113
CALCULATE PRIORITY
RETURN

FIG. 12

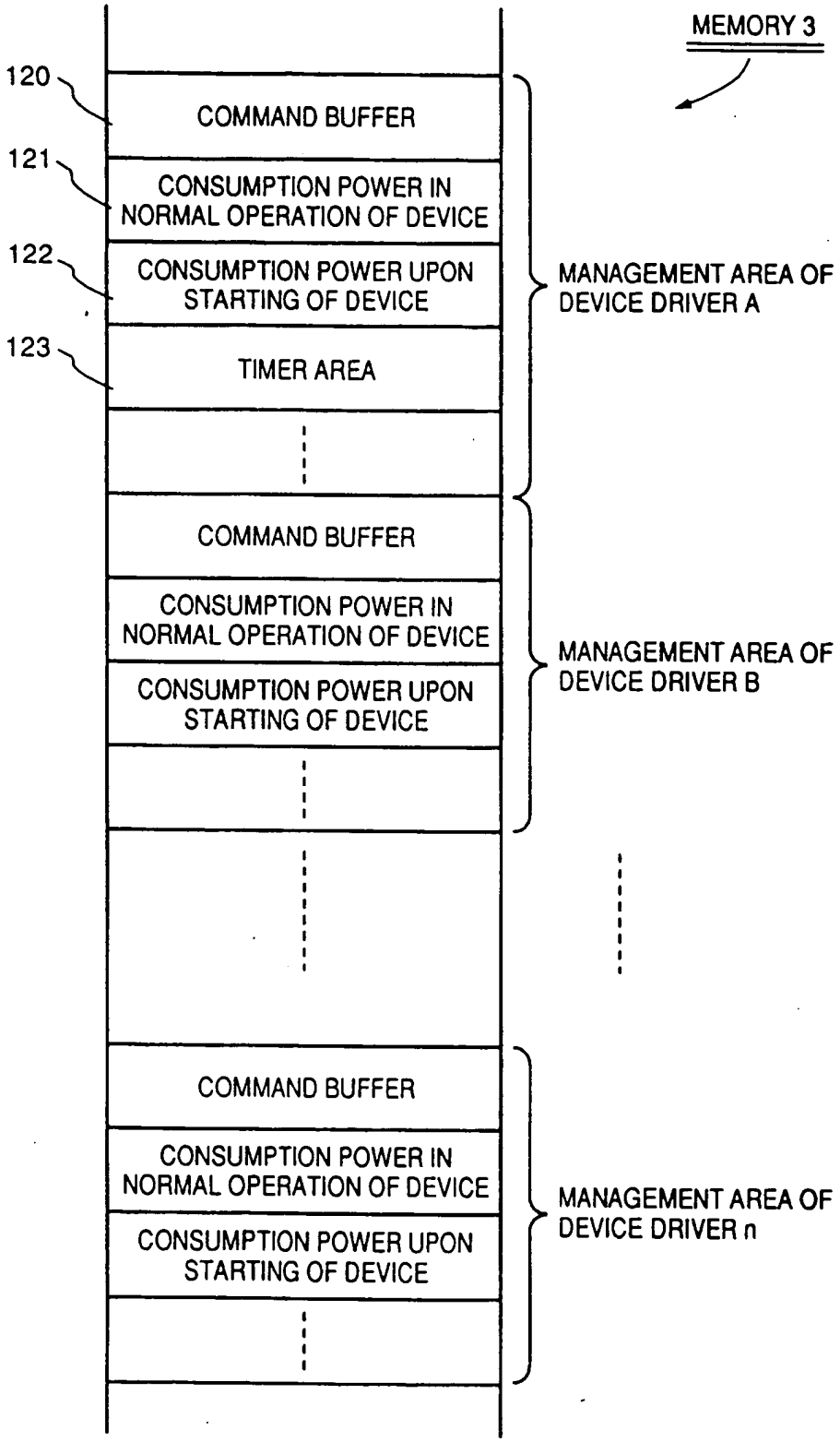

FIG. 14

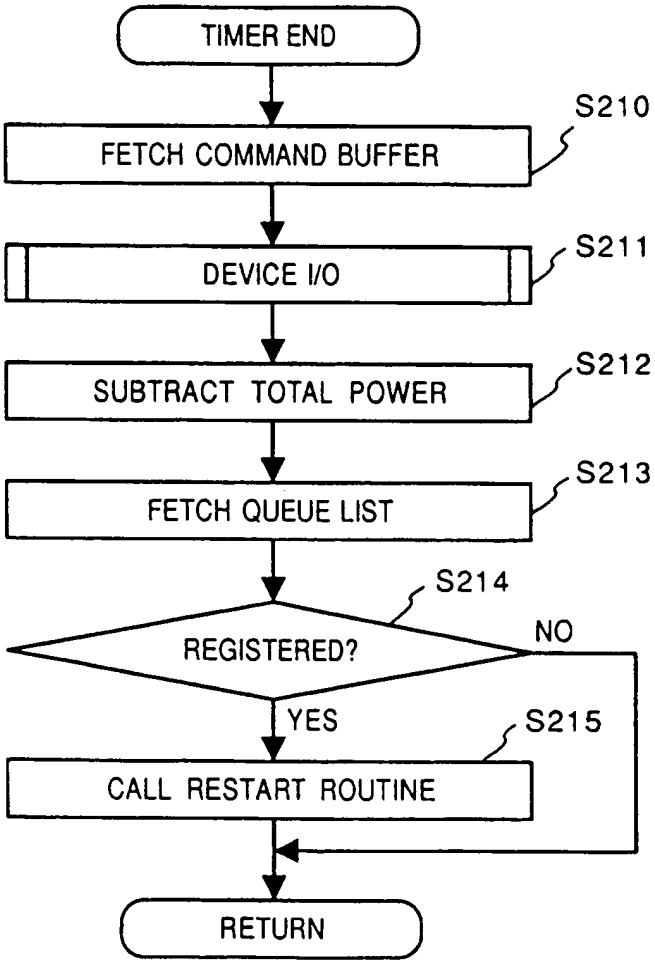

FIG. 16

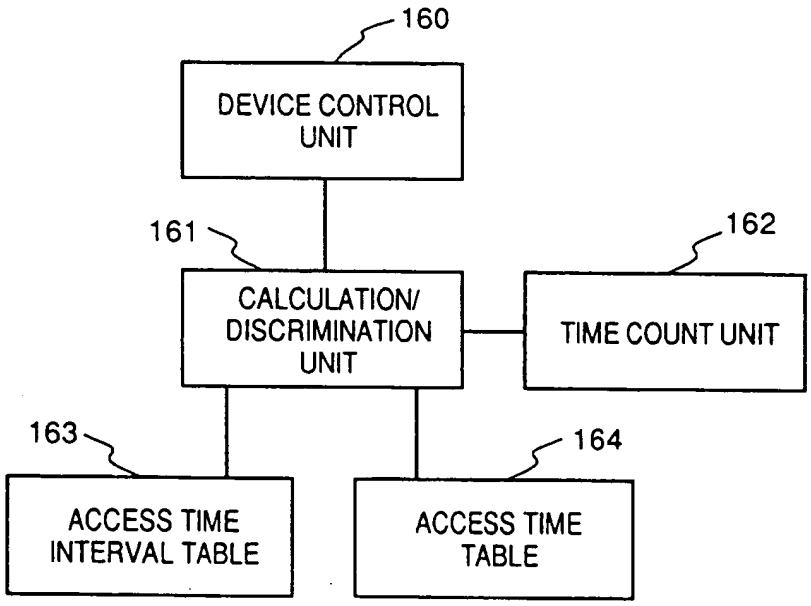

FIG. 18

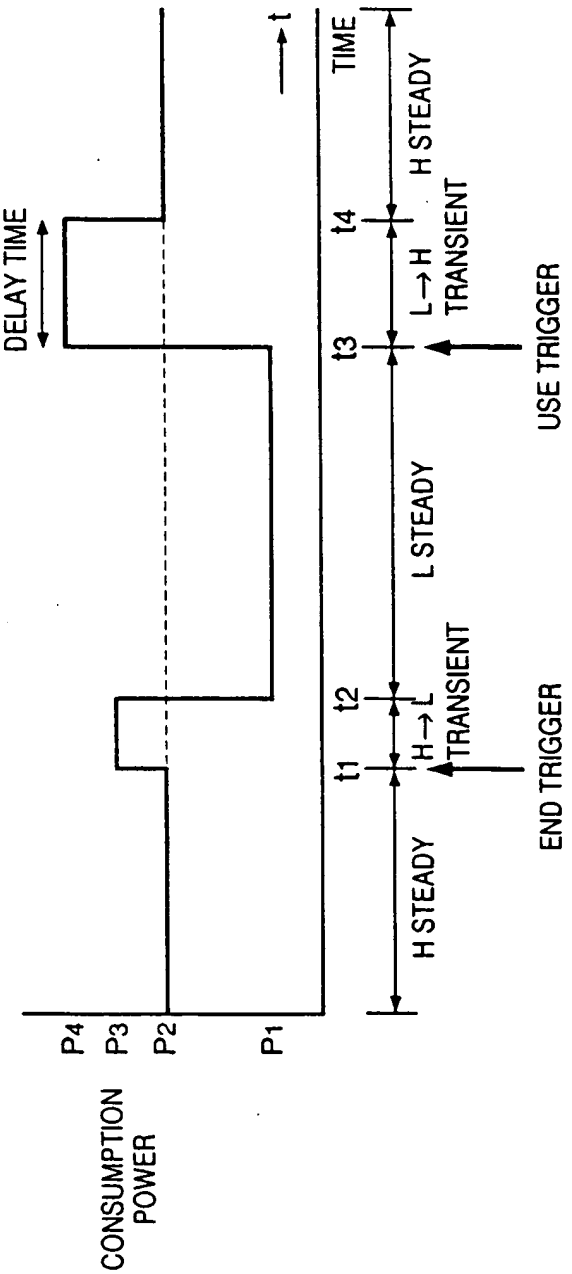

FIG. 20

TIME INTERVAL OF ACCESS TO DEVICE BY APPLICATION (SECONDS)

APPLICATION \ DEVICE	INPUT	OUTPUT	ROM	RAM	HDD	COMMUNICATION
A	5	1	1	1	20	3600
B	120	10	1	1	30	3600
C	60	1	1	1	30	10
D	180	60	30	30	300	60
E	300	360	60	60	1800	3600

FIG. 22

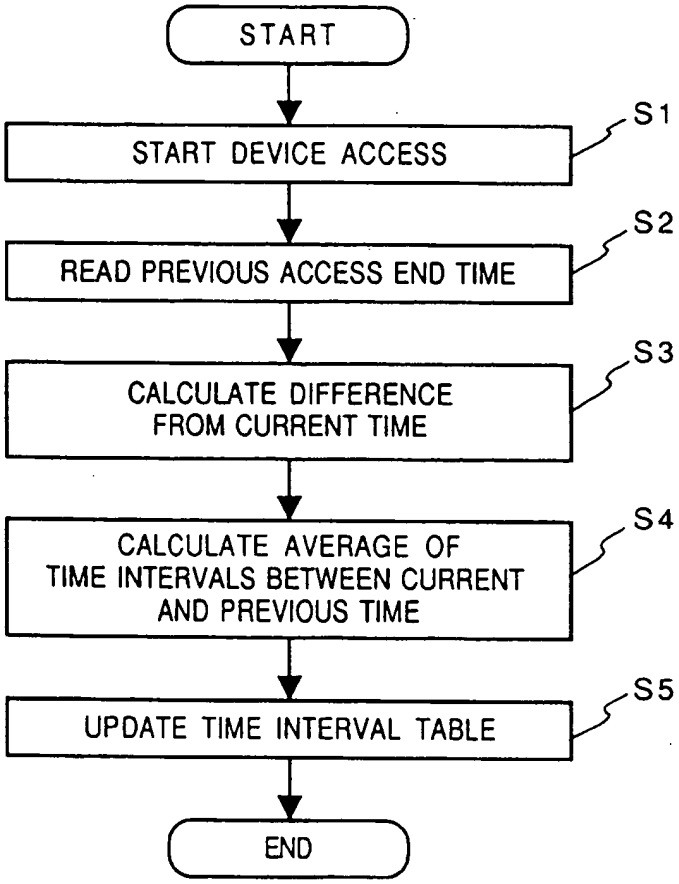

FIG. 24

HISTORY OF t3 - t2
WHEN APPLICATION A ACCESS HDD

ACCESS COUNT	t3 -t2 (SECONDS)
1	22
2	12
3	17
4	24
5	21
6	6
7	13
8	23
9	8

FIG. 25

ACCESS TIME INTERVAL OF HDD (SECONDS)

		FILE	
		OPEN	CLOSE
VIRTUAL MEMORY	ON	6	20
	OFF	10	300

(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 0 683 451 A3

(12) EUROPEAN PATENT APPLICATION

(88) Date of publication A3:
16.04.1997 Bulletin 1997/16

(43) Date of publication A2:
22.11.1995 Bulletin 1995/47

(21) Application number: **95106933.5**

(22) Date of filing: **08.05.1995**

(51) Int. Cl.[6]: **G06F 1/32**, G06F 9/46

(84) Designated Contracting States:
DE FR GB IT NL

(30) Priority: **09.05.1994 JP 95186/94**
20.06.1994 JP 136953/94

(71) Applicant: **CANON KABUSHIKI KAISHA**
Tokyo (JP)

(72) Inventors:
- **Sunakawa, Shinichi,**
  c/o Canon Kabushiki Kaisha
  Tokyo (JP)
- **Shimada, Kazutoshi,**
  c/o Canon Kabushiki Kaisha
  Tokyo (JP)
- **Tatsumi, Eisaku,**
  c/o Canon Kabushiki Kaisha
  Tokyo (JP)
- **Mori, Shigeki,**
  c/o Canon Kabushiki Kaisha
  Tokyo (JP)
- **Matsubayashi, Kazuhiro,**
  c/o Canon Kabushiki Kaisha
  Tokyo (JP)
- **Harada, Takashi,**
  c/o Canon Kabushiki Kaisha
  Tokyo (JP)
- **Nagasaki, Katsuhiko,**
  c/o Canon Kabushiki Kaisha
  Tokyo (JP)
- **Fukuda, Ryoji,**
  c/o Canon Kabushiki Kaisha
  Tokyo (JP)

(74) Representative: **Tiedtke, Harro, Dipl.-Ing. et al**
Patentanwaltsbüro
Tiedtke-Bühling-Kinne & Partner
Bavariaring 4
80336 München (DE)

(54) Power supply control method in multi-task environment

(57) An information processing apparatus, which operates in a multi-task mode, calculates a total consumption power of devices used by each task, and assigns higher execution priority to a task having the largest consumption power, thereby shortening the execution time of the task having the largest consumption power, and suppressing the total consumption power of the apparatus. When a device is started upon switching of tasks, if the total consumption power exceeds the allowable power of the apparatus by a power consumed upon restarting of the device, the task is set in a waiting state until operations of other devices are completed, the consumption power is lowered, and it is ready to use the device by the task.

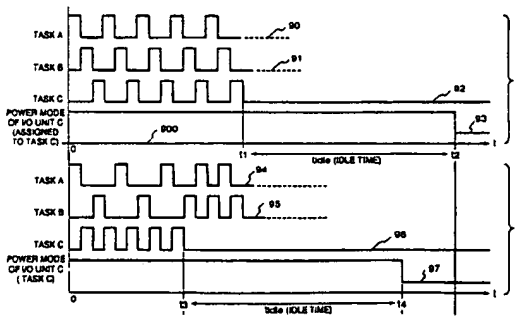

FIG. 5A

FIG. 5B

EP 0 683 451 A3